FSA Integration Partner

United States Department of Education
Federal Student Aid

eZ-Audit - PEPS Integration Clearinghouse File

Task Order #116

March 20, 2003

Revision History

Date	Name (Lotus Notes ID)	Description of Revision
03/21/2003	Ann M. Keast	Initial Creation
04/07/2003	Jennifer C. Sri	Text Changes

Table of Contents

Introduction

Purpose

This document provides a high-level summary and detailed technical design for a component of the eZ-Audit's technical architecture: the Clearinghouse file. The data provided on a weekly basis by the Clearinghouse is critical to the accuracy of the data within the eZ-Audit system.

Scope

The Clearinghouse file is a part of the larger technical architecture framework that will be briefly described. However, the scope of this document will primarily consist of the Clearinghouse file.

Intended audience

This document is intended for eZ-Audit operations team application programmers who need to understand the Clearinghouse file in order to load Federal Audit Clearinghouse data (FAC ACNs, FSA Receipt Date).

Background

In the past, PEPS has been the sole owner and distributor of school and submission data. With the advent of eZ-Audit, some data is still owned by PEPS and consequently eZ-Audit requires setup and interval data from PEPS. However, some data will now be owned by eZ-Audit and PEPS will transfer this data from the eZ-Audit database into their own database.

Historically, the Clearinghouse file was processed by Sherry Quade, an FSA Core Team member, and was entered into the PEPS database. After its "go-live" date, eZ-Audit will receive the Clearinghouse file from Sherry Quade, and the eZ-Audit Operations Team will import the data into the eZ-Audit's database using a batch job. PEPS will then get the Clearinghouse data from eZ-Audit in its Interval Feed. This Clearinghouse file import into eZ-Audit's database will occur on a weekly basis.

Overview

The Clearinghouse file will be processed by the eZ-Audit Operation Team by a weekly batch job. The Clearinghouse file contains several pieces of data, but only a few are significant to eZ-Audit. The few critical pieces of data used by eZ-Audit include: FAC ACN, OPE ID, FSA Receipt Date, and the Audit Period End Date (which is usually the school's fiscal year end).

Design Considerations

Assumptions and Dependencies

It is assumed that this batch job will function in a J2EE application server environment. As the current production server for SFA is IBM WebSphere 3.5.3, the framework will be compiled using its required

JDK version 1.2.2. This batch job should be run weekly, however, it has still not been "scheduled". It is assumed that the ITA logging framework is in the classpath of the batch job at run time. See the References section for ITA Logging Framework details.

General Constraints

The Clearinghouse batch job requires a Comma Separated Value (CSV) file from the Federal Audit Clearinghouse. The batch job expects the file to be in the CSV format from Excel and not the flat file (has a file extension of .SFA) that comes directly from the Clearinghouse. It is assumed the Sherry Quade will still handle this conversion from .sfa to .csv.

Goals and Guidelines

The goal of this development is to provide a simple batch job to validate the Clearinghouse data and add valid FAC ACNs to eZ-Audit's database submission table.

Development Methods

This batch job was developed using general object-oriented software development techniques as are specified in any standard text on the Java programming language. As the framework itself is fairly straightforward in its class and relationship patterns it employs, no object oriented modeling tool or methodology was specifically used in its design.

Detailed Design

The following is the detailed design for the program that will handle reading the file and updating the eZ-Audit database tables accordingly:

Clearinghouse File Upload

Open designated Clearinghouse file for reading
Open error log files for writing errors to be emailed

- Error 1: OPE ID not in eZ-Audit (Institution table)
- Error 2: Duplicate FAC ACN found
- Error 3: FAC ACN value already exists for submission
- Error 4: Submission not yet received in eZ-Audit (OPE ID not in submission table)
- Error 5: OPE ID missing in FAC file

Read each line of file in – storing the follow values in the array

```
Array[count][0] = FAC ACN
Array[count][1] = OPE ID
Array[count][2] = FSA Receipt Date
Array[count][3] = Fiscal Year End
Array[count][4] = FAC Receipt Date
```

Clearinghouse File Upload

For each record do the following:

```
if OPE ID from FAC file is blank
        write to error log - no OPE ID (Error #5)
else
    if OPE ID exists in eZ-Audit submission table
            if FAC ACN is unique
                if FAC ACN exists for OPE ID
                    write to error log - already has an FAC ACN (Error #3)
```

```
                                        else
                                                write FAC ACN to corresponding record
                                        end if
                                else
                                        write to error log - FAC ACN is not unique (Error #2)
                                end if
                        else
                                if OPE ID exists in eZ-Audit institution table
                                        write to error log - no submission yet (Error #4)
                                else
                                        not in eZ-Audit - check why (Error #1)
                                end if
                        end if
        end if
```

System Architecture

The Clearinghouse batch job is a part of the larger technical architecture framework of eZ-Audit's data. The ownership and transfer details of eZ-Audit's school and submission data will not be discussed in this document.

Subsystem Architecture: ITA Email Framework

The ITA Email Framework has been added to the batch job to send a weekly email to an FSA Core Team representative. This framework requires that the framework have access to a Simple Mail Transport Protocol Server (SMTP). SMTP is the protocol that most Internet vendors implement to send emails across the Internet. The SMTP server is the actually workhorse that the Email Framework will connect to and forward emails that applications produce. Once the SMTP server has received the Email, it will connect to its forwarding partner (Another SMTP Server) and forward the Email out to the Internet. Currently the Email framework uses the SMTP Server that exists on the WebSphere machine that is provided via Solaris 2.6.

SubSystem Architecture: ITA Logging and Exception Handling Framework

The ITA Logging and Exception handling frameworks has been added to the batch job to enhance debugging and tracing abilities. This should benefit operations ability to isolate problems that may occur within the batch job. The Logging XML configuration document should be called rcs.xml and located in the home directory of the user making the java calls.

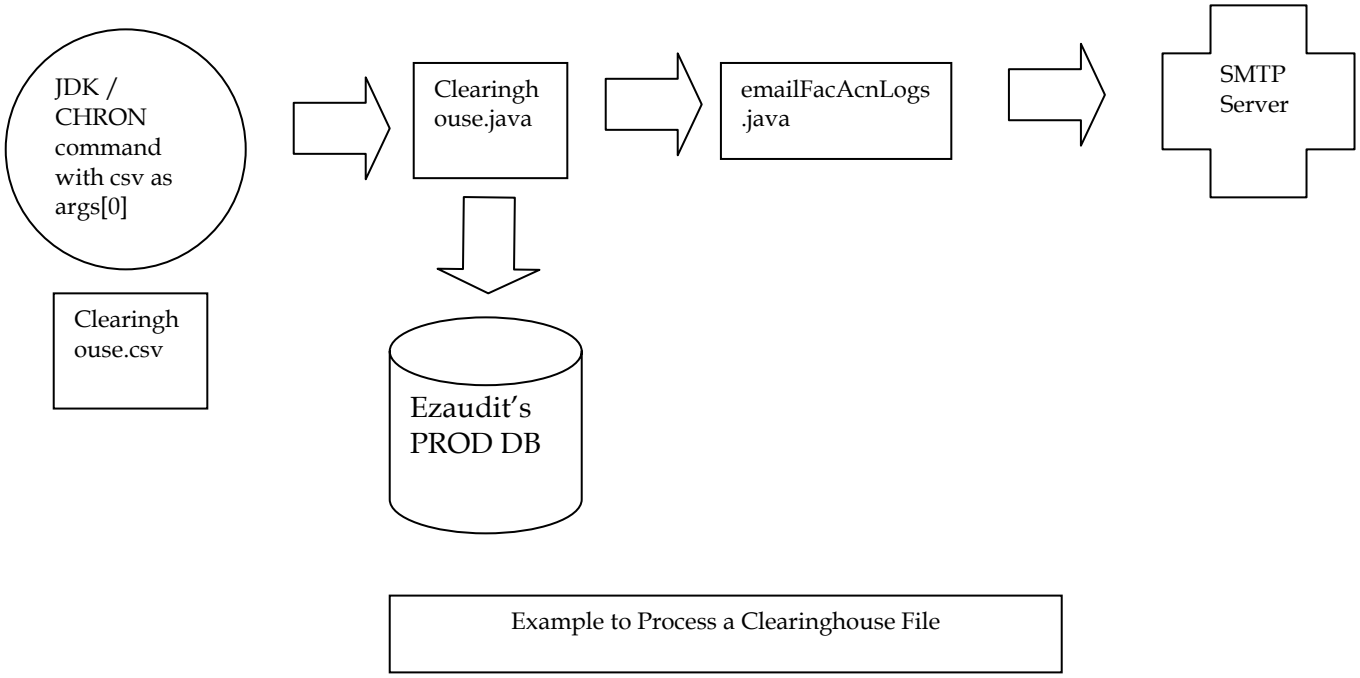
Detailed System Design

clearingHouse Java Class

The clearingHouse class is the program written by the eZ-Audit team, which should run from a VDC application server (development or production). The batch job reads the Clearinghouse file and updates the eZ-Audit database with the appropriate FAC ACNs.

emailFacAcnLogs Java Class

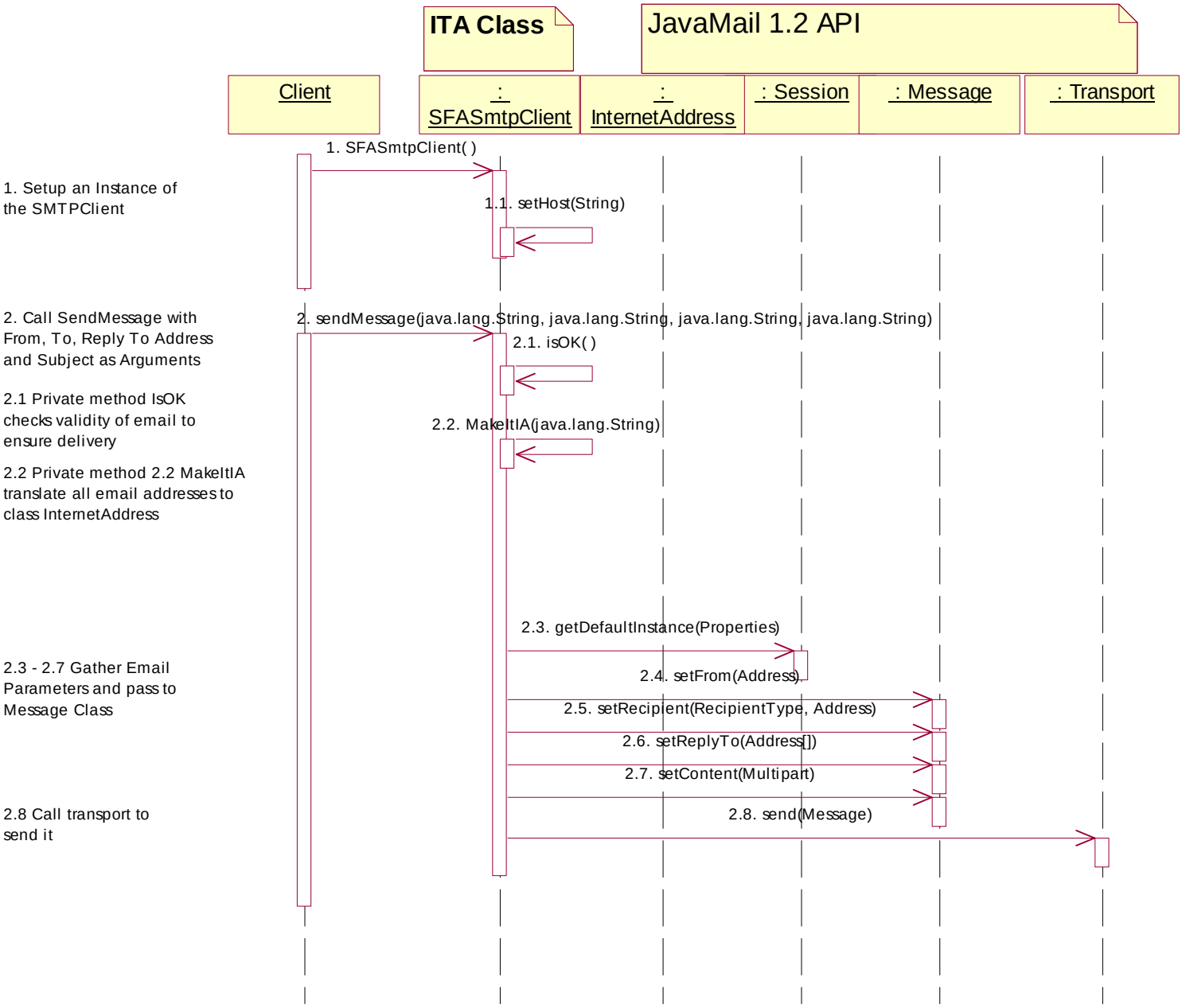
This class uses ITA's email framework to attach the error logs to an email and send it to the FSA Core Team.



Example to Process a Clearinghouse File

Interaction Diagrams

This sequence illustrates the interaction between a client object and SFASmtpClient class to build and send a email to a SMTP Server



References

Java 2 Platform Standard Edition v1.2.2 API Specification
http://java.sun.com/products/jdk/1.2/docs/api/index.html

ITA's Logging Framework Documentation
Z:/ita/documenation/logging/

Appendices

Usage Examples

Jar Files

The following Jars files are part of the ITA Logging Framework.

Logging Jar Files

Jakarta-oro-2.0.1
Jdom-B6
Protomatter-1-1-5.jar
Utility.jar
Xerces.jar
Xml.jar

Logging XML Document

Rcs.xml